МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное учреждение высшего образования

**«Московский технический университет связи и информатики»**

Лабораторная работа №7

**Работа с классами ч.3**

**по дисциплине**

**«Введение в информационные технологии»**

Выполнил: студент гр. БВТ2403

Лашков Ю.Е.

Проверил: Кузнецов В.А.

Москва, 2024 г.

**Цель работы**: Получить практический опыт работы с ООП в Python. использование инкапсуляции, наследования.

**Оборудование:** Компьютер с установленным программным обеспечением Python.

**Введение:** Объектно-ориентированное программирование (ООП) в Python позволяет создавать структурированный и повторно используемый код, что упрощает процесс разработки и сопровождения программного обеспечения.

**Методы и технологии:** В процессе работы применяются такие подходы, как:

- Создание классов для представления объектов

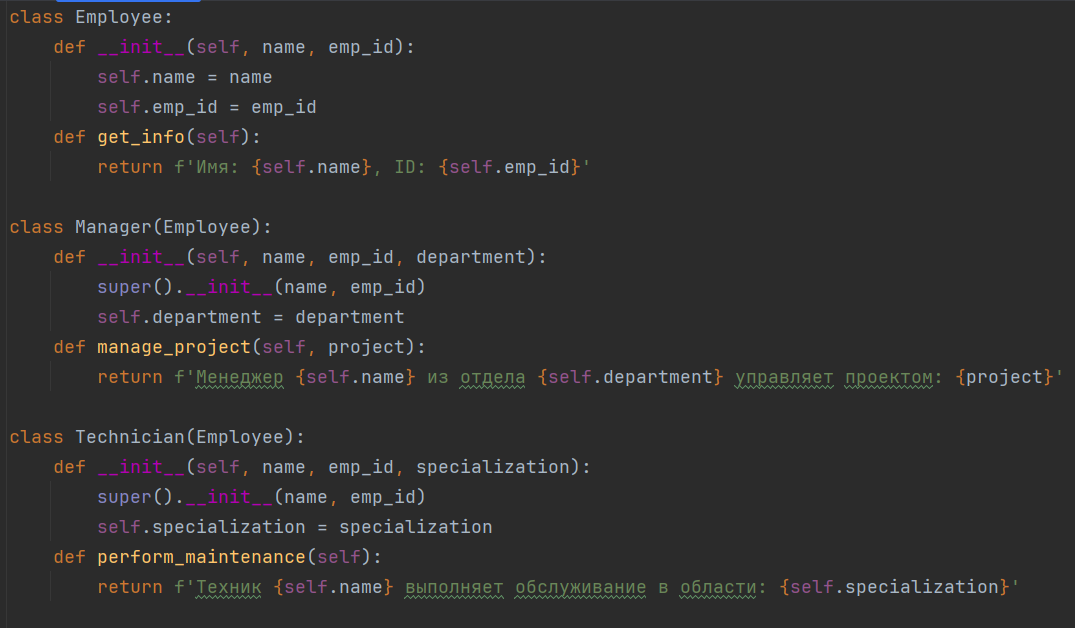
- Инициализация объектов с использованием метода \_\_init\_\_.

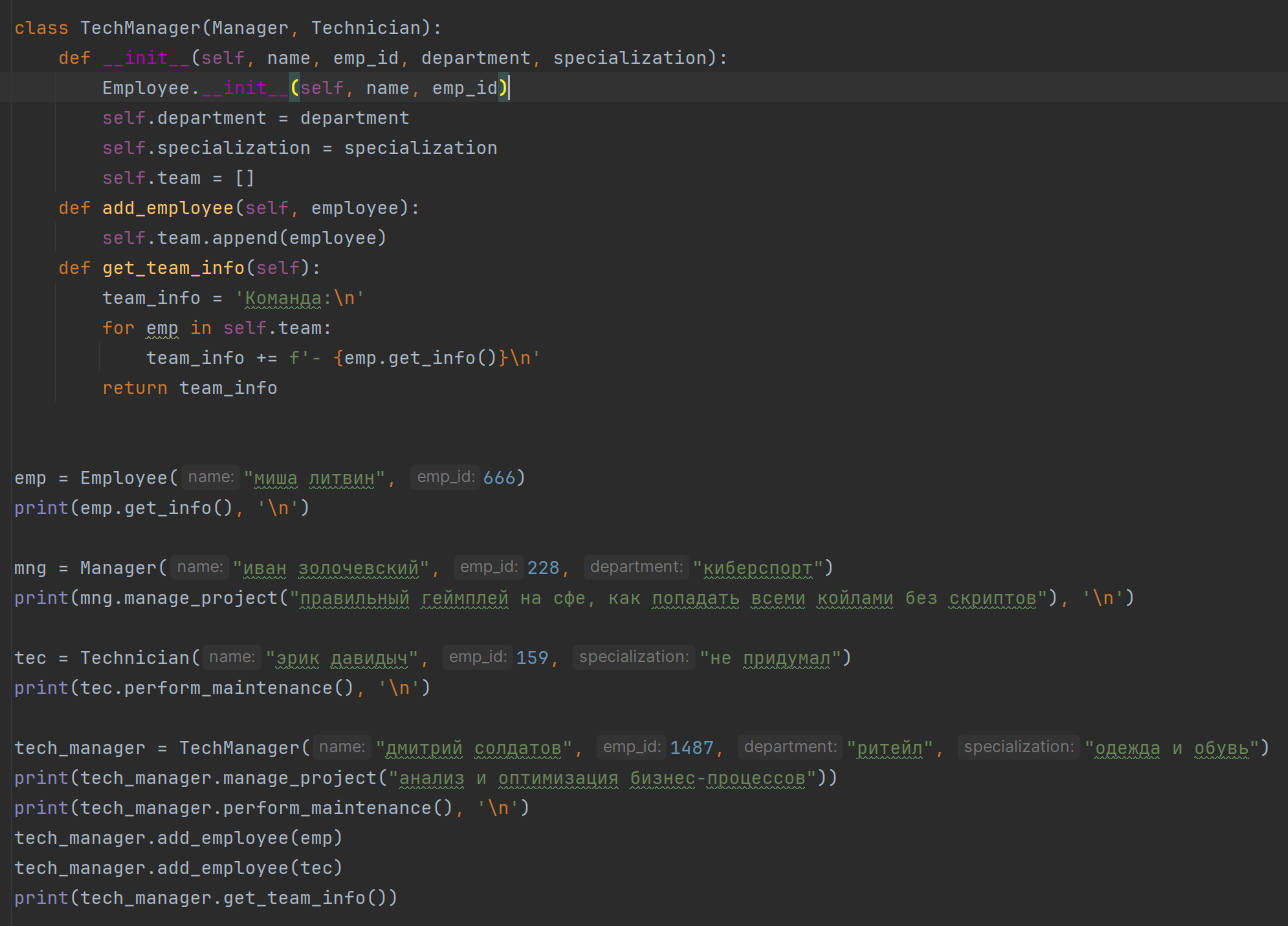
- Работа с атрибутами для хранения и изменения данных внутри объектов.

- Определение методов классов для получения и изменения информации об объектах.

- Использование встроенных функций для вывода информации на экран.

***Задача:***





Код создает иерархию классов для управления сотрудниками в компании, включая базовый класс Employee, который хранит имя и идентификационный номер сотрудника, и предоставляет метод для получения информации. От него наследуются класс Manager, добавляющий атрибут отдела и метод для управления проектами, и класс Technician, который включает специализацию и метод для выполнения обслуживания. Класс TechManager, наследующий как от Manager, так и от Technician, позволяет управлять командой сотрудников, добавляя методы для добавления сотрудников и получения информации о команде. В коде создаются экземпляры этих классов, и вызываются различные методы, демонстрируя их функциональность.

**Заключение:** В ходе работы мы изучили наследование и полиморфизм в Python, а также инкапсуляцию для обеспечения безопасности данных.